图书情报工作
第 63 卷 第 11 期 2019 年 6 月

基于序列到序列模型的抽象式中文文本摘要研究*

余传明[1] 朱星宇[1] 龚雨田[1] 安璐[2]
[1] 中南财经政法大学信息与安全工程学院 武汉 430073 [2] 武汉大学信息管理学院 武汉 430072

摘要：［目的/意义］为更好地处理文本摘要任务中的未登录词（out of vocabulary，OOV），同时避免摘要重复，提高文本摘要的质量，本文以解决 OOV 问题和摘要自我重复问题为研究任务，进行抽象式中文文本摘要研究。［方法/过程］在序列到序列（sequence to sequence，seq2seq）模型的基础上增加指向生成机制和覆盖处理机制，通过指向生成将未登录词拷贝到摘要中以解决未登录词问题，通过覆盖处理避免注意力机制（attention mechanism）反复关注同一位置，以解决重复问题。将本文方法应用到 LCSTS 中文摘要数据集上进行实验，检验模型效果。［结果/结论］实验结果显示，该模型生成摘要的 ROUGE（recall-oriented understudy for gisting evaluation）分数高于传统的 seq2seq 模型以及抽取式文本摘要模型，表明指向生成和覆盖机制能够有效解决未登录词问题和摘要重复问题，从而显著提升文本摘要质量。

关键词：抽象式文本摘要 序列到序列模型 注意力机制 覆盖机制 指向生成机制
分类号：TP391
DOI：10.13266/j.issn.0252-3116.2019.11.012

1 引言

随着大数据时代的高速发展，网络新闻、评论等文本数据呈指数增长，人工生成摘要面临巨大的资源和效率难题，如何利用机器和程序自动对文本进行摘要，通过消除非关键和冗余的信息来压缩并提取文本的主要信息成为研究热点。

依照研究任务的不同，文本摘要可分为抽取式摘要（extractive summarization）和抽象式摘要（abstractive summarization）。前者直接从源文本中抽取具有代表性的文本要素（包括单词、短语和句子）以形成摘要；后者涉及句子的压缩与重构，通过获取源文本的语义表示，利用自然语言生成技术产生摘要。抽象式方法基于文本的语义信息以生成高度抽象的摘要，其结果与源文本在语义上更相似，并且便于用户理解。在抽象式摘要任务中，序列到序列[1]（sequence to sequence，seq2seq）模型是一种较常用的方法。该模型通常基于循环神经网络（recurrent neural network，RNN）对源文档进行编码和解码，其摘要结果具有连贯性，且与源文档的语义相关性较高。在当前研究中，seq2seq 模型仍然面临一些问题。首先，seq2seq 模型无法很好地处理未登录词，从而导致生成的摘要遗漏重要信息。其原因在于，在序列模型中，解码器在每个时间步都会生成一个词语，该词语通常来自于一个固定的词汇表，通过计算概率（例如 softmax 方法）得到。从计算成本和模型训练速度的角度考虑，词汇表通常不会包含训练集中的所有词语，因此部分词语在生成摘要时无法被使用。在训练集中出现的大量低频词语在摘要中会以 UNK（unknown words）形式来表示，较大程度影响摘要的可读性。其次，seq2seq 模型在解码过程中，通常引入注意力机制来改变关注焦点，容易在不同的时间步多次关注同一词语，使得生成的摘要中存在重复片段，降低摘要质量。

为解决上述问题，本文尝试将指向生成机制应用到 seq2seq 模型中，开展中文抽象式文本摘要的实证研究，检验模型效果，以期为相关研究提供借鉴。

* 本文系国家自然科学基金面上项目“大数据环境下基于领域知识获取与对齐的观点检索研究”（项目编号：71373286）和教育部哲学社会科学研究重大课题攻关项目“提高反恐怖主义情报信息工作能力对策研究”（项目编号：17JZD034）研究成果之一。
作者简介：余传明（ORCID：0000-0001-7099-0853），教授；朱星宇（ORCID：0000-0001-8122-3000），硕士研究生；龚雨田（ORCID：0000-0002-0434-2492），硕士研究生；安璐（ORCID：0000-0002-5408-7135），教授，博士生导师，通讯作者，E-mail：anlu97@163.com。
收稿日期：2018-06-15 修回日期：2018-12-16 本文起止页码：108-117 本文责任编辑：王传清

2 相关研究

2.1 抽象式文本摘要

抽象式文本摘要具有良好的连贯性和高凝聚性，近年来成为自然语言处理领域的研究热点。研究者们将多种技术应用于抽象式文本摘要，包括基于结构的方法、基于语义的方法和基于深度学习的方法等。

2.1.1 基于结构的抽象式文本摘要 基于结构的抽象式文本摘要方法主要通过框架、模板、树等模式对文档的重要信息进行编码。例如，H. T. Le 等[2]基于源文本序列、关键词以及句法约束进行句子缩减，利用词图完成句子融合，最终产生包含完整源文档信息且句法正确的抽象摘要。赵文娟等[3]依据相应规则将与事件相关联的信息填充到给定的事件模板中，并以“德国之翼坠机事件”为例验证了该方法的有效性。基于结构的方法较易实现，但其依赖于源文档的篇章结构和形式，在实际应用中具有局限性。

2.1.2 基于语义的抽象式文本摘要 基于语义的抽象式文本摘要方法主要通过自然语言处理技术识别源文档中的名词和动词短语，使用标注和聚类技术确定重要的文档信息，最终将得到的语义表示应用到自然语言生成系统中以生成最终摘要。例如，张晗等[4]基于源文档概念及其语义关系构建语义图，利用语义图中的关键信息生成摘要，结果表明该方法能够有效获取文档的重要信息，生成摘要的准确率、召回率和 F 值较高。A. Khan 等[5]使用语义角色标注识别句子的语义结构，利用改进的图排序算法对重要的图结点进行排序，选择排序最高的图结点生成摘要，在 DUC 数据集上的 ROUGE-1、ROUGE-2 分数分别为 0.417 和 0.108，高于基线方法，显示了该方法的优越性能。王振超等[6]基于文档的语义信息提出一种以事件作为基本语义单元的抽象式摘要方法，对事件进行聚类并利用事件指导摘要语句的生成。基于语义的方法能够很好地捕获源文档的语义信息，能够有效提升摘要与源文档的语义相关性。其局限性在于未使用神经网络自动学习文本特征及表示，模型无法自动学习和生成，因此效率不高。

2.1.3 基于深度学习的抽象式文本摘要 基于深度学习的文本摘要方法通常将文本摘要看作序列到序列的问题，即将源文档作为输入序列，生成的摘要是输出序列。该类方法利用深层次网络，能够更有效学习文本表示，捕获源文档中的重要信息。D. Bahdanau 等[7]最早将 seq2seq 模型用于神经机器翻译任务，利用循环神经网络将源文档编码成固定长度的向量，再解码生成对应翻译。相对于统计学方法，seq2seq 模型具有更好地非线性数据处理能力，但无法很好地处理较长的输入序列，并且对齐效果较差。为进一步改进其效果，A. M. Rush 等[8]在编码器－解码器框架基础上增加注意力机制(attention mechanism)，使解码器在每一时间步关注不同的输入部分，结构化地选取输入子集以降低数据维度，同时使模型更专注于找到与输入数据和当前输出显著相关的有用信息。随后，研究者利用循环解码器[9]、层次网络[10]以及自编码器[11]等对该模型进行改进，进一步提升了模型的效果。谢鸣元等[12]将文档类别信息加入到抽象式摘要中，利用卷积神经网络(convolutional neural network, CNN)对文档进行分类，在 seq2seq 基础上结合文本类别特征生成摘要，相对于传统 seq2seq 模型取得了更高的 ROUGE 分数。

2.2 seq2seq 模型

由于深度学习能够很好地揭示和获取文本信息的内在语义表示，在抽象式文本摘要任务中取得了更好的效果，因此 seq2seq 模型逐渐成为主流。然而在当前研究中，seq2seq 仍面临未登录词问题、重复词汇问题等众多挑战。

2.2.1 未登录词问题 seq2seq 模型通常在训练时会构建一个固定的词表，解码器从该词表中采样来生成词语。考虑到计算效率，研究者通常会根据词频来限制解码词表的规模，导致某些低频词语无法被解码出来，造成未登录词问题。为解决未登录词问题，研究者提出增加解码词表规模、降低词表粒度和采用拷贝机制等方法。

增加解码词表规模最为直接。这类方法专注于提高 softmax 层的处理速度，使词表能够最大限度地包含更多的词汇，从而降低未登录词的出现概率。例如，S. Jean 等[13]利用重要性采样降低在计算输出词语概率相关范数时的复杂度，从而提高解码效率，该方法能够在不显著增加模型复杂度的条件下，构建一个具有更大词汇量的词表。尽管词表的词汇数量足够大且包含训练集中的所有低频词语，但理论上仍不能涵盖所有的词语，因此模型在测试集上的效果无法显著提升。

另一种可行思路是从理论上降低词表的粒度。例如，Z. Xie 等[14]以字母作为编码器－解码器模型的基本处理单元，将其应用于自然语言纠错任务，在 CoNLL 2014 Challenge 数据集上取得了最优的 F0.5 值，很好地解决了未登录词问题。该类方法将模型的输入和输出从以词语为基本单元转变为以字母或字节为基本单元，能够减少未登录词的出现，在一定程度上解决未登

录词问题,但是这种方法会增加模型处理序列的长度从而增大模型的训练难度。

第三种方法是采用拷贝机制。例如,M. T. Luong等[15]利用上下文信息指向未登录词在源文档中的位置,从而将其复制到目标语句中。但是该模型没有使用注意力机制,且模型指向源文档的位置在一个特定的范围内,无法适用于更一般的文本生成任务。J. Gu等[16]提出拷贝网络(CopyNet)模型,该模型将拷贝机制融入到seq2seq模型中,将源文档中的未登录词拷贝到最终摘要中以解决未登录词问题。R. Nallapati等[17]在解码器上配置一个开关,该开关本质是线性层的sigmoid激活函数。开关打开时,解码器按照传统seq2seq的方式从词汇表中生成词语;若开关关闭,解码器指向源文档中的对应位置,并将该位置的词语复制到摘要中。该方法相对于前两类方法效率更高,能够更好地解决未登录词问题。

2.2.2 重复词汇问题 seq2seq模型在解码过程中,通常引入注意力机制来改变关注焦点。注意力机制容易在不同的时间步多次关注同一词语,导致解码器在多个时间步的输入相同,因此最终生成的摘要中存在重复片段。覆盖机制能够很好解决这种重复词汇问题,该机制最早应用于神经机器翻译(neural machine translation,NM)任务,典型的编码器-解码器框架缺乏对已翻译源词语的关注,可能导致过翻译(over translation)和欠翻译(under translation)问题。Z. Tu等[18]在NMT模型中加入覆盖向量来增加对历史注意力的关注,每次注意力更新之后,利用门控神经单元(gated recurrent unit)对该向量进行更新,同时该向量用于调整未来的注意力分布。

在上述背景下,本文尝试将指向生成机制应用到seq2seq模型中:一方面,在seq2seq模型的基础上增加指向生成机制,来处理未登录词。当解码器中的词语是未登录词时,模型指向源文档中该词语的位置,并将对应词语复制到最终摘要中,确保最终摘要的准确性。反之,若解码器中的词语不是未登录词,此时模型与传统序列模型相似,解码器从词汇表中生成新的词语以形成摘要,保持seq2seq模型的抽象生成能力。另一方面,在指向生成器网络基础上增加覆盖机制,避免注意力机制重复关注相同位置,从而减少摘要中的重复词汇。在此基础上,开展中文抽象式文本摘要的实证研究,检验模型效果。

3 研究方法

3.1 研究问题及相关定义

本文的研究任务为抽象式中文文本摘要。假定模型输入一个长度为T的序列 $X = \{x_1, \cdots, x_T\}$,话语生成是指利用序列X和一定的模型,生成长度为M的序列 $Y = \{y_1, \cdots, y_M\}$,其中X为输入的句子序列,Y为输出的句子序列,T和M分别为输入序列和输出序列的长度,且 $T \gg M$。这里的模型由编码器和解码器两部分构成。编码器将序列X按不同时刻输入到编码器,得到其编码 h_t;解码器将该编码 h_t 输入到解码器,从而得到输出序列Y。模型每次输入源文本中一个词,通过词向量层将其转换为分布式表示。

对于给定的源文档Wi,模型的目标是生成由词语y构成的摘要序列,从概率论的角度来看,一般的seq2seq模型在每个时间步会选择概率最大的词语以形成摘要。为简化书写,下文对模型的描述使用表1中的符号。

表1 符号说明

符号	说明
i, t	下标i表示源文档和输入序列中的词语;下标t表示某一时刻
h_t, s_t	分别表示编码器隐藏状态序列和解码器隐藏状态
a_t	在时间步t时的注意力分布
c_t	上下文向量
Pvocab	固定词汇表中所有词语的概率分布
P(y)	生成词语y的概率分布

3.2 模型描述

本文所采用的seq2seq模型架构见图1。它在传统的seq2seq模型[19]基础上添加指向生成机制与覆盖处理机制。在每一时间步,模型通过计算生成概率来决定从源文本中复制词语还是从词汇表中生成词语,利用词汇表中的词语分布和注意力分布得到最终的摘要词语的概率。模型包括3个部分:①编码器、解码器和注意力模块(参见图1中A部分)。在该模块,编码器读取源文档作为输入,得到编码器隐状态,解码器根据编码器隐状态生成解码器隐状态,基于两种隐状态计算每一时间步的注意力分布,得到上下文向量。②指向生成模块(参见图1中C部分)。在该模块,一方面,模型基于上下文向量和解码器隐状态得到词汇表词语分布和生成概率Pgen。另一方面,模型从注意力分布中采样来复制词语,复制概率为(1-Pgen)。基于两部分分布得到目标词语的最终分布,图1词语最终分布中的实心部分来自注意力分布,空心部分来自词汇

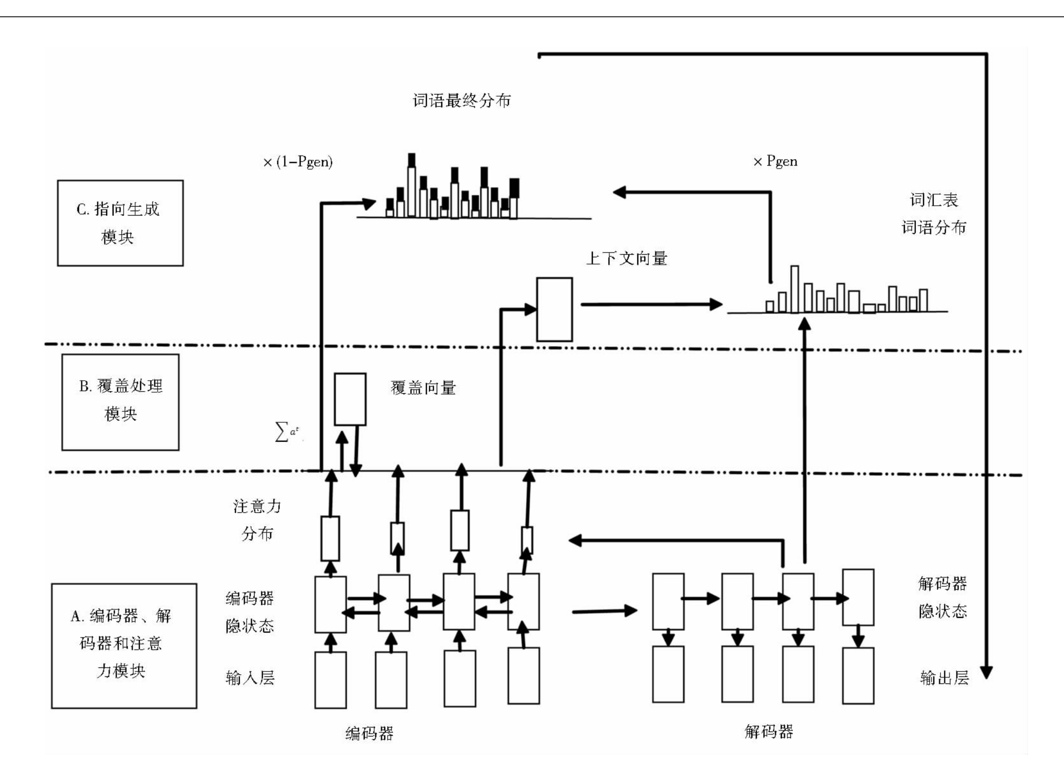


图1 seq2seq模型结构

表词语分布。③覆盖处理模块(参见图1中B部分)。覆盖处理模块对之前时间步的注意力计算加权和,得到覆盖向量,将其作为计算注意力分布的一项额外输入。以下分节展开论述。

3.2.1 编码器、解码器与注意力模块

(1)编码器。编码器由单层双向的长短期记忆网络[20](long short-term memory, LSTM)构成,编码器依次读取输入序列X,在某一时刻t得到的隐状态可由公式(1)计算:

$$h_t = f(x_t, h_{t-1}) \qquad \text{公式(1)}$$

其中,h_{t-1}表示在上一个时刻t－1时的编码器隐状态,x_t是当前时刻输入,f()是一个非线性函数。

基本的seq2seq模型能够按照任意顺序生成词汇表中的词语进而得到最终摘要,而注意力机制能够有效获取源文本序列中的每个词向量,同时确定与输出摘要更加相关的向量,使得模型更加专注于有用的词语。在每一个时间步t,模型根据公式(2)计算当前的注意力分布:

$$a^t = softmax(w^T \tanh(w_1 h_t + w_2 s_t + b_1)) \qquad \text{公式(2)}$$

公式(2)中的w、w_1、w_2以及b1是能够通过训练学习得到的参数。h_t是由公式(1)得到的编码器隐状态,s_t是解码器隐状态。注意力分布可以看作是源文本中词语的概率分布,能够告诉解码器在生成下一个词语时应该关注哪里,概率高的词语在下一时间戳生成摘要词语时会得到更多的关注,从而模型可以生成更能反映源文本信息的词语。

编码器能够将输入序列X通过隐状态序列转化为一个向量,c_t称作上下文向量,如公式(3)计算:

$$c_t = a^t h_t \qquad \text{公式(3)}$$

其中,a^t是由公式(2)得到的在时刻t的注意力分布,作为编码器隐状态h_t的权重,上下文向量能够看作在时刻t学习到的源序列信息的表示,是解码器的输入。

(2)解码器。解码器由单层单向的LSTM构成,根据上下文向量c_t和解码器隐状态s_t产生目标序列Y,如公式(4)预测词表中词语概率分布,模型生成的目标词语概率P(y)与之相同:

$$p_{vocab}(y_t \mid y_{<t}, X) = softmax(y_{t-1}, s_t, c_t) \qquad \text{公式(4)}$$

其中,y_t和y_{t-1}分别是时刻t和t－1时的目标词语,$y_{<t}$表示时刻t之前得到的所有词语,即$\{y_1, \cdots, y_{t-1}\}$,X是输入序列。解码器隐状态s_t可由公式(5)计算:

$$s_t = f(y_{t-1}, s_{t-1}, c_t) \qquad \text{公式(5)}$$

seq2seq模型利用其编码器、解码器和注意力模块计算从固定词表中生成每个目标词语的概率分布,在每个时间步从词表中选择概率最高的词语形成摘要。

3.2.2 指向生成模块 在seq2seq模型中,仅依靠注

意力机制并不能有效处理未登录词，因此增加词语复制机制。受 J. Gu 等[16]和 A. See 等[21]工作启发，本文采用指向生成模块来完成词语复制。在 3.2.1 中已经计算出了注意力分布 a^t 和上下文向量 c_t，基于得到的上下文向量 c_t、解码器状态 s_t 以及解码器输入 x_t，可以利用公式(6)计算词语生成概率 p′，这个概率表示在每个时间步模型从词汇表中生成一个词语作为摘要的可能性：

$$p' = g(w_3 c_t + w_4 s_t + w_5 x_t + b_2) \qquad 公式(6)$$

在公式(6)中，w_3、w_4、w_5 和 b_2 是可训练得到的参数，$g(\,)$ 是一个 sigmoid 激活函数。将 p′看作是一个控制开关，它可以决定模型是从给定的词汇表中生成还是从源文档中复制词语。实际上，这是对源序列中的词语是否是未登录词的一个判断，即如果在该时间步解码器的目标词语是未登录词，则 p′值很小，模型从源文档中复制词语以生成摘要；否则，模型将基于词汇表生成新的词语。基于此，可以得到目标词语 y 概率分布可由公式(7)计算：

$$p(y) = p' P_{vocab} + (1 - p') \sum_i a_i^t \qquad 公式(7)$$

在公式(7)中，p′即由公式(6)计算得到的词语生成概率，P_{vocab} 是公式(4)得到的词表中词语的概率分布，$\sum_i a_i^t$ 表示 y 是未登录词时，得到的注意力分布之和。从公式(7)可以看出，如果 y 是一个未登录词，即表明解码器在该时间步生成的词语没有出现在给定的词汇表中，因此在词汇表中的概率 P_{vocab} 为 0，这说明此时生成的词语 y 来自源文档，需要从源文档中复制得到；相反，若 y 没有出现在源文档中，则累计注意力分布值为 0，此时模型从给定的词表中生成词语。能够很好地处理未登录词并将其复制到最终摘要中是本文提出的指向生成器网络的一个主要优点，而基于注意力机制的序列模型则受到预先设定词表的限制不能解决未登录词的生成问题。

3.2.3 覆盖处理模块 本文利用覆盖处理模块来避免摘要片段重复。基于在所有之前的解码器时间步中的注意力分布之和，得到一个覆盖向量，该向量能够告知模型在之前的时间步已经关注过的词语，因此在该时间步不需要重复注意，从而避免重复词语的产生。在 $t = 0$ 时，覆盖向量是一个零向量，因为在第一个时间步上，源文档中的所有词语都还没有被覆盖，注意力分布为 0 导致覆盖向量取 0。覆盖向量是注意力机制的一个额外的输入，将其直接添加到计算注意力分布的公式(2)中即可。这能够保证在使用注意力机制时，当前时间步选择关注的部分受到先前时间步关注部分的影响。因此这能够避免注意力机制在多个时间步反复关注同一个部分，从而避免模型最终生成重复文本，这也正是覆盖机制的核心思想。

4 实验及分析

4.1 数据集

本文采用 B. Hu 等[22]构建的大规模中文短文本摘要数据集(large scale Chinese short text summarization dataset, LCSTS)。该数据集包含从新浪微博上获取的超过 240 万条文本及相应作者给出的摘要，每条文本不少于 80 个字符，对应摘要长度介于 10 个字符与 30 个字符之间。为保证文本质量，研究者收集 50 个受欢迎(具有蓝“V”标志且微博粉丝数量超过 100 万)的组织用户，如人民日报、经济观察报和国防部等作为种子，捕获其发布的微博，这些微博文本涉及政治、经济、军事、电影和游戏领域。原完整数据集包含 3 部分，PART I 包含 2 400 591 个文本摘要对，PART II 和 PART III 分别包含 10 666 和 1 106 个文本摘要对。本文选择其中数据量最多的部分(即 PART I)的数据进行实验。使用中文分词工具 jieba[23]对数据进行分词处理，将分词后的数据处理为二进制文件，并分为 18 个训练集数据文件、1 个验证集数据文件以及 1 个测试集数据文件。另外，seq2seq 模型中生成摘要的固定词汇表文件包含 40 万词语，实验过程中可以通过设置词汇表大小选择实际用来实验的词语。

4.2 评价指标

为评价不同模型生成的摘要质量，本文将 ROUGE[24]分数作为评价指标，该评价指标基于生成摘要和参考摘要(标准摘要)中 n 元词汇(N-Gram)的重叠情况来评价自动生成的摘要结果，是一种面向 n 元词召回率的评价方法。其基本思想是，首先由专家生成人工摘要，构成参考摘要集(标准摘要集)，将模型自动生成的摘要与标准摘要相对比，通过统计两者之间重叠的基本单元的数目来评价不同模型摘要的质量。待评估摘要与标准摘要中匹配的 N 元词语(N 可取 1、2、3 等自然数)越多，ROUGE 分数越高，说明模型生成的摘要越接近标准摘要，因此质量较高，该方法现已成为摘要评价技术的通用指标之一[24]。ROUGE 评价指标由一系列的评价方法组成，包括 ROUGE-N(N 可取 1、2、3 等自然数)和 ROUGE-L 等。其中，ROUGE-1 和 ROUGE-2 分别代表基于模型生成的摘要与标准摘要之间的 1 元词和 2 元词重叠程度，ROUGE-L 代表基于生成摘要和标准摘要之间的最长公共子序列的重

叠程度。在本文实验中,选取上述指标中的 ROUGE-1、ROUGE-2 和 ROUGE-L 来评价自动生成摘要的质量。

4.3 对比方法

本文针对未登录词以及摘要片段重复问题,在 seq2seq 模型的编码器、解码器和注意力模块上增加指向生成模块和覆盖处理模块。为更好地研究指向生成模块和覆盖处理模块的效果,在实验中通过对参数进行设置,使用基于注意力机制的 seq2seq 模型(Attention),增加指向生成模块的 seq2seq 模型(Attention + PG),以及同时增加指向生成模块与覆盖处理模块的 seq2seq 模型(Attention + PG + Coverage)3 种抽象式方法进行中文文本摘要实验。将使用指向生成模块与覆盖处理模块模型的实验结果与 seq2seq 的实验结果进行对比。

同时,为进一步比较本文方法相对于抽取式摘要方法的效果,使用基于 TextRank[25] 的方法、Lead-1-First(抽取原文本中第一句话)和 Lead-1-Last(抽取原文本中最后一句话)这 3 种典型的抽取式方法作为对照。

4.4 参数设置

本实验中,神经网络隐藏状态为 256 维,源序列与目标序列中词向量都是 128 维,使用包含 50 000 个词语的词汇表。实验没有预先训练词向量,在训练阶段从头开始进行学习,使用初始学习率为 0.15 且初始累加器值为 0.1 的 Adagrad 优化算法进行优化。Adagrad 算法差异化地给每个参数分配学习率,这个过程是自适应进行的。随着参数更新的总距离的增加,其学习速率也随之减慢。在测试阶段,使用束大小为 4 的束搜索(beam search)来产生摘要。

4.5 基础实验结果评价

按照以上设置在 LCSTS 数据集上分别进行摘要实验,利用 pyrouge 包计算得到的 ROUGE 分数结果如表 2 所示:

表 2 不同摘要方法在测试集上的 ROUGE 分数

ROUGE 分数	ROUGE-1	ROUGE-2	ROUGE-L
Lead-1-First	0.111 8	0.033 8	0.103 8
Lead-1-Last	0.134 0	0.045 7	0.123 4
TextRank	0.129 3	0.039 9	0.119 3
Attention	0.105 4	0.009 6	0.101 4
Attention + PG	0.308 3	0.113 6	0.284 3
Attention + PG + Coverage	**0.348 7**	**0.114 7**	**0.306 1**

从表 2 可以看出,在各种序列到序列方法中,具有覆盖机制和指向生成机制的模型(Attention + PG + Coverage)在 3 个指标上都取得了最好的效果,其 ROUGE 分数分别为 0.348 7、0.114 7 以及 0.306 1;仅增加指向生成机制的模型(Attention + PG)生成摘要的效果次于前者(Attention + PG + Coverage),在 ROUGE-1、ROUGE-2 和 ROUGE-L 上的分数分别低 0.040 4、0.001 1 以及 0.021 8;传统的基于注意力机制的 seq2seq 模型在 ROUGE-1、ROUGE-2 以及 ROUGE-L 评价指标上分数都最低,分别为 0.105 4、0.009 6 和 0.101 4,远低于另外两种 seq2seq 模型(Attention + PG 和 Attention + PG + Coverage)的实验结果。在传统抽取式方法中,Lead-1-Last 方法在 ROUGE-1、ROUGE-2 以及 ROUGE-L 评价指标上分数最高,分别为 0.134 0、0.045 7和0.123 4,略高于 TextRank 和 Lead-1-First 抽取式方法。

综合对比抽取式方法和抽象式方法可以看出,具有覆盖机制和指向生成机制的模型在 ROUGE-1 和 ROUGE-L 上都具有更好的效果。相比于 3 种抽取式方法,Attention + PG + Coverage 模型在 ROUGE-1 分数上分别提升 0.236 9、0.214 7 和 0.219 4,在 ROUGE-2 分数上分别提升 0.080 9,0.069 0 和 0.074 8,在 ROUGE-L 分数上分别提升 0.202 3、0.182 7 和 0.113 1。实验结果表明,与传统的抽象式和抽取式模型相比,本文所提出的模型通过结合指向生成机制与覆盖机制,能够有效提升中文文本摘要的效果。

表 3 显示了不同摘要模型在相同新闻文章上生成的摘要比较。可以看出,利用传统 seq2seq 模型(Attention)生成的摘要中包含较多重复的词语片段,即源文本的某些细节信息被错误地反复生成,且这些重复的片段通常由在训练集中出现较频繁的词语组成,而频度较低的词语(仍然包含在词汇表中)往往会被更常见的词语代替。例如在表 3 中,该方法生成的摘要中“被骗”重复出现了 3 次,这种重复较大程度上降低了摘要的可读性。另外,摘要中的“深圳”显然与源文本中的“天津”一词不符,摘要无法准确反映源文本信息。通过查阅训练数据生成的固定词汇表,可以看出,“深圳”一词在训练集中出现的频度为 49 398 次,而“天津”一词在训练集中出现的频度为 15 212 次,相比之下“深圳”一词更为常见。因此,基线方法在训练时更容易学习到“深圳”的向量表示,而学习到的“天津”的向量表示较弱,最终导致从词汇表中生成摘要时更容易生成错误的常见词语。除此之外,基线方法生成的摘要中存在多个[UNK]表示,表明传统的序列模型无法生成未能包含在词汇表中的 OOV 词语,损失了源文档的重要信息,无法生成包含源文档全部信息以及

语义完整的摘要,生成的摘要质量不高。

在加入指向生成机制后(Attention + PG),可以看到对于同一个源文档,利用指向生成器网络产生的摘要中,则将对应的[UNK]表示替换为从源文档中复制得到的命名实体等内容,因此最终摘要的可读性更高,且几乎包含了源文档中的重要信息。这表明,利用指向生成机制能够很好地处理 OOV 词语。尽管使用指向生成机制产生的摘要消除了[UNK]标识,然而生成的摘要中"天津警方破获犯罪嫌疑人"这一片段重复出现,造成了生成摘要的冗余。

由表 3 可以看出,在同时加入指向生成机制和覆盖机制后(Attention + PG + Coverage),生成的摘要形式良好,且包含源文档的重要信息。摘要结果中未出现不能被模型识别的[UNK]标记,同时消除了摘要中的重复片段,最终得到的摘要与参考摘要在内容和语义上更加一致,与摘要结果更加相符。

表 3 3 种抽象摘要方法基于同一篇新闻产生的摘要比较

来源和方法	结果
源文本	目前,天津警方破获了一起特大假冒箱包案,一实体店长期销售、批发假冒路易威登、古驰、香奈儿、巴宝莉等品牌箱包,警方查获各类品牌包袋 7000 余个,按正品估价值上亿元! 目前涉嫌售假的犯罪嫌疑人已被抓获。
参考摘要	天津破获特大假冒箱包案,LV、香奈儿均被仿冒涉值上亿。
Attention	深圳警方破获特大售假案宣判:[UNK][UNK][UNK][UNK]被骗 1 个月被骗 1 亿! [UNK][UNK]被骗!! [UNK]!!!!!!
Attention + PG	天津警方破获特大假冒箱包案警方查 7000 余个价值上亿元——目前涉嫌售假的犯罪嫌疑人已被抓获——天津警方破获犯罪嫌疑人
Attention + PG + Coverage	天津查获各类品牌包袋 7000 余个犯罪嫌疑人已被抓获。按正品估价值上亿!

4.6 扩展实验结果评价

本文的深层神经网络涉及词表大小、词向量维度等多个参数,实验中设置神经网络中词向量维度为 128 维,词汇表包含 50 000 个词语。为进一步研究不同的超参数对模型产生摘要质量的影响,在 Attention + PG + Coverage 模型上分别设置不同的词表大小和词向量维度进行摘要实验,比较模型在不同的词表大小和词向量维度设置下的 ROUGE 分数,以进一步检验模型效果。

4.6.1 词表大小对于实验结果的影响 表 4 显示本文提出的模型在使用不同的词汇表进行横向比较实验时,生成摘要得到的 ROUGE 分数比较情况。从表 4 可以看出,随着词汇表数量从 20 000 增加到 80 000,模型生成摘要的 ROUGE 分数整体呈现先增加后减少的趋势。在词汇表大小为 60 000 时,模型的 ROUGE-1 和 ROUGE-L 分数最高,分别为 0.358 1 和 0.314 8。当词汇表包含 70 000 词汇时,对应的 ROUGE-2 分数最高,为 0.117 8。特别地,当词汇表大小为 40 000 时,模型的 ROUGE 分数低于词汇表大小为 30 000 时,分别降低 0.017 2、0.006 8 和 0.018 4。实验结果表明,Attention + PG + Coverage 模型在不同的词汇表大小设置下具有不同的实验效果,词汇表大小对模型效果具有一定的影响,当词汇表为特定数量(本文为 60 000)时,模型具有最佳的效果,生成的摘要质量较好。

4.6.2 向量维度对于实验结果的影响 表 5 显示本文提出的模型在设置不同的词向量维度进行横向比较实验时, 生成摘要得到的 ROUGE 分数比较。可以看出,当词向量维度为 128 维时,模型效果最好,相对于使用 64 维词向量的模型,ROUGE 分数分别提高0.018 4、0.003 4 以及 0.012 3,与使用 128 维词向量的模型相比,ROUGE 分数分别提升 0.023 8、0.003 7 和 0.015 8。从表 5 中实验结果可以看出,词向量维度对 Attention + PG + Coverage 模型效果具有一定的影响,在设置词向量维度为 128 维时,模型生成的摘要质量最好。

表 4 使用不同大小词表的方法得到的 ROUGE 分数比较

Attention + PG + Coverage	ROUGE-1	ROUGE-2	ROUGE-L
20 000 词汇	0.318 8	0.105 4	0.285 0
30 000 词汇	0.339 1	0.115 7	0.304 8
40 000 词汇	0.321 9	0.108 9	0.286 4
50 000 词汇	0.348 7	0.114 7	0.306 1
60 000 词汇	**0.358 1**	0.115 6	**0.314 8**
70 000 词汇	0.350 0	**0.117 8**	0.310 2
80 000 词汇	0.320 0	0.108 8	0.287 0

表 5 使用不同向量维度的方法得到的 ROUGE 分数比较

Attention + PG + Coverage	ROUGE-1	ROUGE-2	ROUGE-L
64 维词向量	0.330 3	0.111 3	0.293 8
128 维词向量	**0.348 7**	**0.114 7**	**0.306 1**
256 维词向量	0.324 9	0.111 0	0.290 3

4.7 讨论

4.7.1 模型的总体实验效果分析 从模型的总体实验效果来看,与传统的基于注意力机制的 seq2seq 模型相比,本文所提出的 Attention + PG + Coverage 模型能够更好地处理未登录词语以及重复词汇问题,从而有效提升抽象式中文文本摘要的效果。

在处理未登录词方面,传统的 seq2seq 模型在最终生成的摘要中将其用[UNK]标记来代替,这些标记与

参考摘要内容无法匹配,导致最终的词语匹配率降低,进而使得其 ROUGE 分数较低。与传统的 seq2seq 模型相比,在实验参数相同的情况下,增加指向生成网络的模型能够显著提升 ROUGE 值。这表明,通过指向源文档中的词语并将其复制到摘要中,能够更好地解决抽象式文本摘要中的未登录词问题,从而提升摘要质量。

在处理重复词汇问题方面,传统的 seq2seq 模型采用注意力机制,在不同时间步重复地关注源文档相同位置,从而生成相同的摘要片段。而在大多数情况下,这些重复片段与参考摘要并不吻合,从而导致 ROUGE 分数较低。在 seq2seq 模型的基础上引入指向生成网络能够更好地处理未登录词问题,却由于生成更多不必要的重复片段,而使得摘要变得冗余。在指向生成网络的基础上增加覆盖机制,能够有效消除指向生成网络所带来的词汇重复内容,因此取得了比指向生成网络更好的效果。

4.7.2　参数设置对模型的影响分析　从参数设置对模型的影响来看,在从词汇表大小和词向量维度两方面对 Attention + PG + Coverage 模型进行横向对照实验时,词汇表大小以及词向量维度在较大程度上影响到模型在抽象式中文文本摘要上的效果。

对词汇表大小而言,当词汇表在特定规模(例如 60 000)时,既能包含足够的高频词语,又能排除一定的低频词语,在从词汇表中采样时能够生成较多源文档信息的词语作为摘要。且对于未能包含在词汇表中的词语,通过指向生成模块将其拷贝到摘要中,因此生成的摘要质量最好。当词汇表规模较小(例如 20 000、30 000、40 000 和 50 000)时,训练集中部分频度较高的词语未能包含在词汇表中(与频度最高的词语相比,较高的词语被忽略)。模型在从词汇表中采样时,将概率最高的词语纳入摘要,从而使生成的摘要中只包含较少的高频关键词,与参考摘要的匹配程度较低。当词汇表规模较大(例如 70 000 和 80 000)时,在训练数据中出现频度较低的词语也将包含在词汇表中。这些低频词语在模型学习过程中具有较弱的词向量,即使它们包含源文档重要信息也很难被选择用以形成摘要,因此最终得到的摘要倾向于缺乏重要信息,与参考摘要具有差异。

对词向量维度而言,在设置特定词向量维度(例如 128 维)时,分布式表示既能很好地捕获源文档词语的语义信息,同时能够缓解词向量稀疏,模型效果远优于使用其它向量维度得到的结果,从而生成质量较高的抽象摘要。当词向量维度较小(例如 64 维)时,模型学习到的分布式表示无法很好地捕获源文档中词语的含义,进而模型不能很好地获取源文档的抽象信息,因此生成的摘要与参考摘要存在差距。当词向量维度设置较高(例如 256 维)时,可能导致学习到的词向量相对稀疏,无法很好地表示词语的语义信息和词语间的内在联系,从而使得基于这种向量表示生成的摘要质量较差。

4.7.3　模型的局限性分析　从实验结果与实际生成的摘要对照来看,利用指向生成网络和覆盖机制生成的摘要大多数效果良好,相对于基线方法有较大的提升,但最终生成的摘要存在少数与实验结果不一致的情况。例如,具有覆盖机制的指向生成器生成的摘要“美国研究:肥胖的人出现记忆丧失的可能性高出 3 倍,你知道吗? 你知道吗?”中,仍然存在不必要的重复片段“你知道吗?”。这可能由于该片段在训练数据中出现频度较高,在源文档中也多次出现,因而模型在不同的时间步有较大概率重复关注相同位置,导致摘要出现重复内容。而利用指向生成网络产生的摘要中仍然存在少量的[UNK]标记,如“男子喝 8 两白酒去江边游泳,[UNK] 10 小时”,笔者分析原因是存在一些在训练集中出现频度较低但仍包含在词汇表中的词语(出现频度排序处于 TOP K 的词语),其向量表示较弱,在模型学习过程中无法从词汇表中准确生成,同时模型计算出的词语复制概率较低,因此未能从源文档中复制,最终以[UNK]标记代替。

5　结语

为解决抽象式文本摘要中的未登录词问题和摘要重复问题,本文在 seq2seq 模型的编码器、解码器和注意力模块上增加指向生成模块和覆盖处理模块。从实验结果来看,一方面,模型能够将指向源文档中的未登录词拷贝到最终摘要中,从而很好地解决传统序列模型中普遍存在的未登录词问题,消除摘要中的[UNK]标记。另一方面,覆盖处理模块能够避免模型在每一时间步反复地关注源文档的相同位置,进而避免生成重复的摘要片段。实验结果表明,在抽象式中文文本摘要任务中,利用指向生成网络和覆盖机制能够有效解决未登录词问题和摘要重复问题,从而显著提升文本摘要质量。

本文的不足之处在于:①实验数据集局限于中文,在后续工作中,将采用更多语言的数据集对 seq2seq 模型用于抽象式文本摘要进行研究;②仅与

部分经典的非 seq2seq 模型方法(包括 TextRank、Lead-1-First 以及 Lead-1-Last)进行对比,在后续工作中,将与更多的非序列模型进行比较,以进一步验证模型的有效性。

参考文献:

[1] SUTSKEVER I, VINYALS O, LE Q V. Sequence to sequence learning with neural networks[C]// Proceedings of 2014 annual conference on neural information processing systems (NIPS). Montreal: Neural Information Processing Systems Foudation, 2014: 3104-3112.

[2] LE H T, LE T M. An approach to abstractive text summarization[C]//Proceedings of 2013 soft computing and pattern recognition (SoCPaR). Hanoi: IEEE, 2013: 371-376.

[3] 赵文娟, 刘忠宝. 基于汉语框架的网络事件抽取及相关算法研究[J]. 情报理论与实践, 2016, 39(10):112-116.

[4] 张晗, 赵玉虹. 基于语义图的医学多文档摘要提取模型构建[J]. 图书情报工作, 2017, 61(8):112-119.

[5] KHAN A, SALIM N, FARMAN H, et al. Abstractive text summarization based on improved semantic graph approach[J]. International journal of parallel programming, 2018, 46(1):1-25.

[6] 王振超, 孙锐, 姬东鸿. 基于事件指导的多文档生成式摘要方法[J]. 计算机应用研究, 2017, 34(2):343-346.

[7] BAHDANAU D, CHO K, BENGIO Y. Neural machine translation by jointly learning to align and translate[EB/OL]. [2017-12-30]. https://arxiv.org/pdf/1409.0473.pdf.

[8] RUSH A M, CHOPRA S, WESTON J. A neural attention model for abstractive sentence summarization [EB/OL]. [2017-12-30]. https://arxiv.org/pdf/1509.00685.

[9] CHOPRA S, AULI M, RUSH A M. Abstractive sentence summarization with attentive recurrent neural networks[C]// Conference of the North American chapter of the Association for Computational Linguistics. San Diego: Human Language Technologies, 2016:93-98.

[10] GULCEHRE C, AHH S, NALLAPATI R, et al. Pointing the unknown words[C]// Proceedings of the 54th annual meeting of the Association for Computational Linguistics. Berlin: ACL, 2016:140-149.

[11] MIAO Y, BLUNSOM P. Language as a latent variable: discrete generative models for sentence compression[C]// Proceedings of the 2016 conference on empirical methods in natural language processing. Austin: EMNLP, 2016:319-328.

[12] 谢鸣元. 基于文本类别的文本自动摘要模型[J]. 电脑知识与技术: 学术交流, 2018, 14(1): 206-208.

[13] JEAN S, CHO K, MEMISEVIC R, et al. On using very large target vocabulary for neural machine translation [EB/OL]. [2018-02-10]. https://arxiv.org/pdf/1412.2007.pdf.

[14] XIE Z, AVATI A, ARIVAZHAGAN N, et al. Neural language correction with character-based attention [EB/OL]. [2017-12-30]. https://arxiv.org/pdf/1603.09727.

[15] LUONG M T, SUTSKEVER I, LE Q V, et al. Addressing the rare word problem in neural machine translation[J]. Bulletin of university of agricultural sciences and veterinary medicine cluj-napoca. veterinary medicine, 2014, 27(2):82-86.

[16] GU J, LU Z, LI H, et al. Incorporating copying mechanism in sequence-to-sequence learning [C]//Proceedings of the 54th annual meeting of the Association for Computational Linguistics. Berlin: ACL, 2016:1631-1640.

[17] NALLAPAT R, ZHOU B, SANTOS C N D, et al. Abstractive text summarization using sequence-to-sequence RNNs and beyond [C]// Proceedings of the 20th SIGNLL conference on computational natural language learning. Berlin: CoNLL, 2016:280-290.

[18] TU Z, LU Z, LIU Y, et al. Modeling coverage for neural machine translation[C]// Proceedings of the 54th annual meeting of the Association for Computational Linguistics. Berlin: ACL, 2016:76-85.

[19] CHO K, MERRIENBOER B V, GULCEHRE C, et al. Learning phrase representations using RNN encoder-decoder for statistical machine translation [EB/OL]. [2018-03-01]. https://arxiv.org/pdf/1406.1078.pdf

[20] HOCHREITER S, SCHMIDHUBER J. Long Short-Term Memory [J]. Neural computation, 1997, 9(8):1735-1780.

[21] SEE A, LIU P J, MANNING C D. Get to the point: summarization with pointer-generator networks [C]//Proceedings of the 55th annual meeting of the Association for Computational Linguistics. Vancouver: ACL, 2017:1073-1083.

[22] HU B, CHEN Q, ZHU F. LCSTS: a large scale Chinese short text summarization dataset [C]// Proceedings of the 2015 conference on empirical methods in natural language processing. Lisbon: EMNLP, 2015:2667-2671.

[23] SUN J. 中文分词工具 [EB/OL]. [2017-10-20]. https://pypi.python.org/pypi/jieba/.

[24] FLICK C. ROUGE: a package for automatic evaluation of summaries [EB/OL]. [2017-12-30]. http://www.aclweb.org/anthology/W04-1013.

[25] MIHALCEA R, TARAU P. TextRank: bringing order into texts [C]// Proceedings of the 2004 conference on empirical methods in natural language processing. Barcelona: EMNLP, 2004: 404-411.

作者贡献说明:

余传明:论文构思、实验数据获取、模型实验、论文初稿撰写与修改;

朱星宇:基线方法和词汇表大小扩展实验、论文初稿撰写与修改;

龚雨田:数据集预处理、向量维度扩展实验、论文修改;

安璐:论文构思与修改。

余传明，朱星宇，龚雨田，等.基于序列到序列模型的抽象式中文文本摘要研究[J].图书情报工作,2019,63(11):108－117.

Research of Abstractive Chinese Text Summarization Based on Seq2seq Model

Yu Chuanming[1] Zhu Xingyu[1] Gong Yutian[1] An Lu[2]
[1] School of Information and Safety Engineering, Zhongnan University of Economics and Law, Wuhan 430073
[2] School of Information Management, Wuhan University, Wuhan 430072

Abstract: [**Purpose/significance**] To deal with the Out Of Vocabulary (OOV) in text summarization while avoiding duplication of summaries, this article focuses on solving the OOV problem and the self-duplication and carries out a profiling study. [**Method/process**] Bases on the sequence-to-sequence model, a pointer generator module and a coverage processing module are added. An attempt is made to copy the OOV into abstractive summary to solve the problem of OOV by means of the pointer generator module. The coverage processing module tries to avoid the Attention Mechanism paying attention to the same position repeatedly to solve the duplicate problem. The model is applied to the Chinese summarization dataset LCSTS to conduct experiments to test the effectiveness. [**Result/conclusion**] Experiment results show that the ROUGE of the generated summary is much higher than that of seq2seq model and extractive model, indicating that in the abstractive Chinese text summary, the pointer generator module and the coverage mechanism module can effectively solve the problem of OOV and the repetition of the summary, thereby significantly improving text summary quality.

Keywords: abstractive text summarization sequence-to-sequence model attention mechanism coverage mechanism pointer generator mechanism